Improving Information Retrieval Performance by Combining Different Text-Mining Techniques

Rila Mandala RILA@INFORMATIKA.ORG
Department of Informatics,
Institute Technology of Bandung,
Jl. Ganesha 10, Bandung 40132,
Indonesia

Takenobu Tokunaga TAKE@CL.CS.TITECH.AC.JP
Hozumi Tanaka TANAKA@CL.CS.TITECH.AC.JP
Department of Computer Science
Tokyo Institute of Technology,
2-12-1 Oookayama Meguro-Ku, Tokyo 152-8554, Japan

Abstract

WordNet, a hand-made, general-purpose, and machine-readable thesaurus, has been used in information retrieval research by many researchers, but failed to improve the performance of their retrieval system. Thereby in this paper we investigate why the use of WordNet has not been successful. Based on this analysis we propose a method of making WordNet more useful in information retrieval applications by combining it with other knowledge resources. A simple word sense disambiguation is performed to avoid misleading expansion terms. Experiments using several standard information retrieval test collections show that our method results in a significant improvement of information retrieval performance. Failure analysis were done on the cases in which the proposed method fail to improve the retrieval effectiveness. We found that queries containing negative statements and multiple aspects might cause problems in the proposed method and we also investigated the solution to these problems.

1. Introduction

Information comes in many forms: news, financial data, scientific research, etc. It represents one of the most important commodities in the modern world. Modern computing and networking technology make it possible to organize, store, and transport large bodies of data with minimal effort anywhere in the world. Without question, we have moved into the information age.

With so much material so easily accessible, many organizations and individuals have realized that the real issue is no longer getting enough information, but selectively pick out what is useful to them from vast quantities of material. Information retrieval systems and software designed to index, store, and provide easy access to data, are rapidly developing to meet this need. As one of their most important features, the systems provide search tools, algorithms which map an expression of the user's information need into a mathematical form which is then used to identify relevant material in the database.

Information retrieval is concerned with locating documents relevant to a user's information needs from a collection of documents. The user describes his/her information needs

with a query which consists of a number of words. The information retrieval system compares the query with documents in the collection and returns the documents that are likely to satisfy the user's information requirements. A fundamental weakness of current information retrieval methods is that the vocabulary that searchers use is often not the same as the one by which the information has been indexed. Query expansion is one method to solve this problem. The query is expanded using terms which have similar meaning or bear some relation to those in the query, increasing the chances of matching words in relevant documents. Expanded terms are generally taken from a thesaurus.

Obviously, given a query, the information retrieval system must present all useful articles to the user. This objective is measured by recall, i.e. the proportion of relevant articles retrieved by the system. Conversely, the information retrieval system must not present any useless article to the user. This criteria is measured by precision, i.e. the proportion of retrieved articles that are relevant.

Development of WordNet began in 1985 at Princeton University (Miller, 1990). A team lead by Prof. George Miller aimed to create a source of lexical knowledge whose organization would reflect some of the recent findings of psycholinguistic research into the human lexicon. WordNet has been used in numerous natural language processing, such as part of speech tagging (Segond, Schiller, Grefenstette, & Chanod, 97), word sense disambiguation (Resnik, 1995a), text categorization (Gomez-Hidalgo & Rodriguez, 1997), information extraction (Chai & Biermann, 1997), and so on with considerable success. However the usefulness of WordNet in information retrieval applications has been debatable.

Two sets of experiments using the TREC collection were performed to investigate the effectiveness of using WordNet for query expansion by Voorhees (Voorhees, 1994). The first set used handpicked synsets and the second set extends the expansion strategy to include automatically selecting the starting synsets. When the concepts were chosen manually, her method could improve the retrieval effectiveness for short queries, but failed to improve the retrieval effectiveness for long queries. When the concepts were chosen automatically, none of the expansion methods produced significant improvement as compared with an unexpanded run. She further tried to use WordNet as a tool for word sense disambiguation (Voorhees, 1993) and applied it to text retrieval, but the performance of retrieval was degraded.

Stairmand (Stairmand, 1997) used WordNet to investigate the computational analysis of lexical cohesion in text using lexical chain method (Morris & Hirst, 1991). Because lexical chains are associated with topics, he suggested that information retrieval, where the notion of topic is very pertinent, is a suitable application domain. He concluded that his method only succeed in small-scale evaluation, but a hybrid approach is required to scale-up to real-word information retrieval scenarios.

Smeaton and Berrut (Smeaton & Berrut, 1995) tried to expand the queries of the TREC-4 collection with various strategies of weighting expansion terms, along with manual and automatic word sense disambiguation techniques. Unfortunately all strategies degraded the retrieval performance.

Instead of matching terms in queries and documents, Richardson (Richardson & Smeaton, 1995) used WordNet to compute the semantic distance between concepts or words and then used this term distance to compute the similarity between a query and a document. Al-

though he proposed two methods to compute semantic distances, neither of them increased the retrieval performance.

2. Limitations of WordNet

In this section we analyze why WordNet has failed to improve information retrieval performance. We ran exact-match retrieval against 9 small standard test collections (Fox, 1990) in order to observe this phenomenon. An information retrieval test collection consists of a collection of documents along with a set of test queries. The set of relevant documents for each test query is also given, so that the performance of the information retrieval system can be measured. We expand queries using a combination of synonyms, hypernyms, and hyponyms in WordNet. The results are shown in Table 1.

In Table 1 we show the name of the test collection (Collection), the total number of documents (#Doc) and queries (#Query), and all relevant documents for all queries (#Rel) in that collection. For each document collection, we indicate the total number of relevant documents retrieved (Rel-ret), the recall ($\frac{\text{Rel-ret}}{\#\text{Rel}}$), the total number of documents retrieved (Ret-docs), and the precision ($\frac{\text{Rel-ret}}{\text{Ret-docs}}$) for each of no expansion (Base), expansion with synonyms (Exp. I), expansion with synonyms and hypernyms (Exp. II), expansion with synonyms and hyponyms (Exp. III), and expansion with synonyms, hypernyms, and hyponyms (Exp. IV).

From the results in Table 1, we can conclude that query expansion can increase recall performance but unfortunately degrades precision performance. We thus turned to investigation of why all the relevant documents could not be retrieved with the query expansion method above. Some of the reasons are stated below :

- Two terms that seem to be interrelated have different parts of speech in WordNet. This is the case between *stochastic* (adjective) and *statistic* (noun). Since words in WordNet are grouped on the basis of part of speech in WordNet, it is not possible to find a relationship between terms with different parts of speech.

- Most of relationships between two terms are not found in WordNet. For example how do we know that Sumitomo Bank is a Japanese company ?

- Some terms are not included in WordNet (proper name, etc).

To overcome all the above problems, we propose a method to enrich WordNet with an automatically constructed thesaurus. The idea underlying this method is that an automatically constructed thesaurus could complement the drawbacks of WordNet. For example, as we stated earlier, proper names and their interrelations among them are not found in WordNet, but if proper names and other terms have some strong relationship, they often co-occur in the document, so that their relationship may be modeled by an automatically constructed thesaurus.

Polysemous words degrade the precision of information retrieval since all senses of the original query term are considered for expansion. To overcome the problem of polysemous words, we apply a restriction in that queries are expanded by adding those terms that are

Table 1: Term Expansion Experiment Results using WordNet

Collection	#Doc	#Query	#Rel		Base	Exp. I	Exp. II	Exp. III	Exp. IV
ADI	82	35	170	Rel-ret	157	159	166	169	169
				Recall	0.9235	0.9353	0.9765	0.9941	0.9941
				Ret-docs	2,063	2,295	2,542	2,737	2,782
				Precision	0.0761	0.0693	0.0653	0.0617	0.0607
CACM	3204	64	796	Rel-ret	738	756	766	773	773
				Recall	0.9271	0.9497	0.9623	0.9711	0.9711
				Ret-docs	67,950	86,552	101,154	109,391	116,001
				Precision	0.0109	0.0087	0.0076	0.0070	0.0067
CISI	1460	112	3114	Rel-ret	2,952	3015	3,076	3,104	3,106
				Recall	0.9479	0.9682	0.9878	0.9968	0.9974
				Ret-docs	87,895	98,844	106,275	108,970	109,674
				Precision	0.0336	0.0305	0.0289	0.0284	0.0283
CRAN	1398	225	1838	Rel-ret	1,769	1,801	1,823	1,815	1,827
				Recall	0.9625	0.9799	0.9918	0.9875	0.9940
				Ret-docs	199,469	247,212	284,026	287,028	301,314
				Precision	0.0089	0.0073	0.0064	0.0063	0.0060
INSPEC	12684	84	2543	Rel-ret	2,508	2,531	2,538	2,536	2,542
				Recall	0.9862	0.9953	0.9980	0.9972	0.9996
				Ret-docs	564,809	735,931	852,056	869,364	912,810
				Precision	0.0044	0.0034	0.0030	0.0029	0.0028
LISA	6004	35	339	Rel-ret	339	339	339	339	339
				Recall	1.0000	1.0000	1.0000	1.0000	1.0000
				Ret-docs	148,547	171,808	184,101	188,289	189,784
				Precision	0.0023	0.0020	0.0018	0.0018	0.0018
MED	1033	30	696	Rel-ret	639	662	670	671	673
				Recall	0.9181	0.9511	0.9626	0.9640	0.9670
				Ret-docs	12,021	16,758	22,316	22,866	25,250
				Precision	0.0532	0.0395	0.0300	0.0293	0.0267
NPL	11429	100	2083	Rel-ret	2,061	2,071	2,073	2,072	2,074
				Recall	0.9894	0.9942	0.9952	0.9942	0.9957
				Ret-docs	267,158	395,280	539,048	577,033	678,828
				Precision	0.0077	0.0052	0.0038	0.0036	0.0031
TIME	423	24	324	Rel-ret	324	324	324	324	324
				Recall	1.000	1.000	1.000	1.000	1.000
				Ret-docs	23,014	29,912	33,650	32,696	34,443
				Precision	0.0141	0.0108	0.0096	0.0095	0.0094

most similar to the entirety of query terms, rather than selecting terms that are similar to a single term in the query.

2.1 Co-occurrence-based Thesaurus

The general idea underlying the use of term co-occurrence data for thesaurus construction is that words that tend to occur together in documents are likely to have similar, or related, meanings (Qiu & Frei, 1993). Co-occurrence data thus provides a statistical method for automatically identifying semantic relationships that are normally contained in a hand-made thesaurus. Suppose two words (A and B) occur f_a and f_b times, respectively, and co-occur f_c times, then the similarity between A and B can be calculated using a similarity coefficient such as the Tanimoto Coefficient :

$$\frac{f_c}{f_a + f_b - f_c}$$

2.2 Syntactically-based Thesaurus

In contrast to the previous section, this method attempts to gather term relations on the basis of linguistic relations and not document co-occurrence statistics. Words appearing in similar grammatical contexts are assumed to be similar, and therefore classified into the same class (Lin, 1998; Grefenstette, 1994, 1992; Ruge, 1992; Hindle, 1990).

First, all the documents are parsed using the Apple Pie Parser. The Apple Pie Parser is a natural language syntactic analyzer developed by Satoshi Sekine at New York University (Sekine & Grishman, 1995). The parser is a bottom-up probabilistic chart parser which finds the parse tree with the best score by way of the best-first search algorithm. Its grammar is a semi-context sensitive grammar with two non-terminals and was automatically extracted from Penn Tree Bank syntactically tagged corpus developed at the University of Pennsylvania. The parser generates a syntactic tree in the manner of a Penn Tree Bank bracketing. Figure 1 shows a parse tree produced by this parser.

The main technique used by the parser is the best-first search. Because the grammar is probabilistic, it is enough to find only one parse tree with highest possibility. During the parsing process, the parser keeps the unexpanded active nodes in a heap, and always expands the active node with the best probability.

Unknown words are treated in a special manner. If the tagging phase of the parser finds an unknown word, it uses a list of parts-of-speech defined in the parameter file. This information has been collected from the Wall Street Journal corpus and uses part of the corpus for training and the rest for testing. Also, it has separate lists for such information as special suffices like -ly, -y, -ed, -d, and -s. The accuracy of this parser is reported as parseval recall 77.45 % and parseval precision 75.58 %.

Using the above parser, the following syntactic structures are extracted :

- Subject-Verb
- Verb-Object
- Adjective-Noun

Each noun has a set of verbs and adjective that it occurs with, and for each such relationship, a Tanimoto coefficient value is calculated.

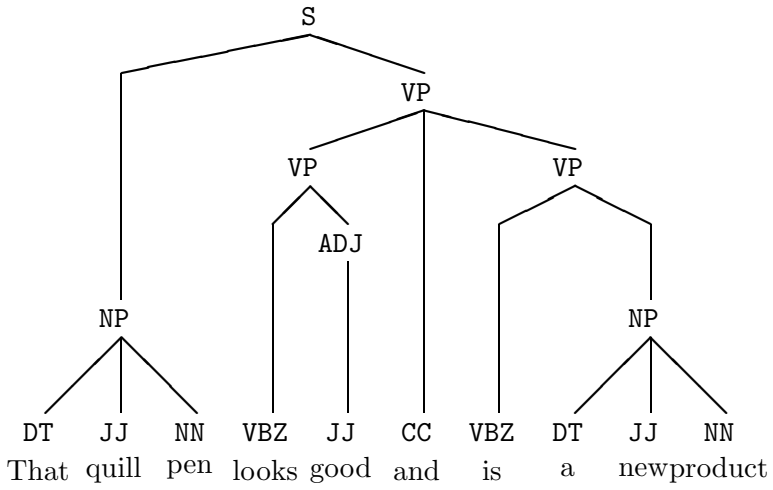

Figure 1: An example parse tree

- $C_{sub}(v_i, n_j) = \frac{f_{sub}(v_i,n_j)}{f(v_i)+f_{sub}(n_j)-f_{sub}(v_i,n_j)}$,
  where $f_{sub}(v_i, n_j)$ is the frequency of noun n_j occurring as the subject of verb v_i, $f_{sub}(n_j)$ is the frequency of the noun n_j occurring as subject of any verb, and $f(v_i)$ is the frequency of the verb v_i

- $C_{obj}(v_i, n_j) = \frac{f_{obj}(v_i,n_j)}{f(v_i)+f_{obj}(n_j)-f_{obj}(v_i,n_j)}$,
  where $f_{obj}(v_i, n_j)$ is the frequency of noun n_j occurring as the object of verb v_i, $f_{obj}(n_j)$ is the frequency of the noun n_j occurring as object of any verb, and $f(v_i)$ is the frequency of the verb v_i

- $C_{adj}(a_i, n_j) = \frac{f_{adj}(a_i,n_j)}{f(a_i)+f_{adj}(n_j)-f_{adj}(a_i,n_j)}$,
  where $f(a_i, n_j)$ is the frequency of noun n_j occurring as argument of adjective a_i, $f_{adj}(n_j)$ is the frequency of the noun n_j occurring as argument of any adjective, and $f(a_i)$ is the frequency of the adjective a_i

We define the similarity of two nouns with respect to one predicate as the minimum of each Tanimoto coefficient with respect to that predicate, i.e.,
$SIM_{sub}(v_i, n_j, n_k)$=$min\{C_{sub}(v_i, n_j), C_{sub}(v_i, n_k)\}$
$SIM_{obj}(v_i, n_j, n_k)$=$min\{C_{obj}(v_i, n_j), C_{obj}(v_i, n_k)\}$
$SIM_{adj}(a_i, n_j, n_k)$=$min\{C_{adj}(a_i, n_j), C_{adj}(a_i, n_k)\}$

Finally the overall similarity between two nouns is defined as the average of all the similarities between those two nouns for all predicate-argument structures.

2.3 Expansion Term Weighting Method

A query q is represented by a vector $\overrightarrow{\mathrm{q}} = (q_1, q_2, ..., q_n)$, where the q_i's are the weights of the search terms t_i contained in query q.

The similarity between a query q and a term t_j can be defined as belows (Qiu & Frei, 1993):

$$simqt(q, t_j) = \sum_{t_i \in q} q_i * sim(t_i, t_j)$$

Where the value of $sim(t_i, t_j)$ can be defined as the average of the similarity values in the three types of thesaurus.

With respect to the query q, all the terms in the collection can now be ranked according to their *simqt*. Expansion terms are terms t_j with high $simqt(q, t_j)$.

The $weight(q, t_j)$ of an expansion term t_j is defined as a function of $simqt(q, t_j)$:

$$weight(q, t_j) = \frac{simqt(q, t_j)}{\sum_{t_i \in q} q_i}$$

where $0 \leq weight(q, t_j) \leq 1$.

An expansion term gets a weight of 1 if its similarity to all the terms in the query is 1. Expansion terms with similarity 0 to all the terms in the query get a weight of 0. The weight of an expansion term depends both on the entire retrieval query and on the similarity between the terms. The weight of an expansion term can be interpreted mathematically as the weighted mean of the similarities between the term t_j and all the query terms. The weight of the original query terms are the weighting factors of those similarities.

Therefore the query q is expanded by adding the following query :

$$\overrightarrow{\mathrm{q_e}} = (a_1, a_2, ..., a_r)$$

where a_j is equal to $weight(q, t_j)$ if t_j belongs to the top r ranked terms. Otherwise a_j is equal to 0.

The resulting expanded query is :

$$\overrightarrow{\mathrm{q}}_{expanded} = \overrightarrow{\mathrm{q}} \circ \overrightarrow{\mathrm{q_e}}$$

where the $\circ$ is defined as the concatenation operator.

The method above can accommodate the polysemous word problem, because an expansion term which is taken from a different sense to the original query term is given very low weight.

3. Experiments

3.1 Evaluation method

Recall and precision are two widely used metrics to measure the retrieval effectiveness of an information retrieval system. Recall is the fraction of the relevant documents which has been retrieved, i.e.

$$recall = \frac{number\ of\ relevant\ documents\ retrieved}{number\ of\ relevant\ documents\ in\ collection}.$$

Precision is the fraction of the retrieved document, i.e.

$$precision = \frac{number\ of\ relevant\ documents\ retrieved}{total\ number\ of\ documents\ retrieved}.$$

However, precision and recall are set-based measures. That is, they evaluate the quality of an unordered set of retrieved documents. To evaluate ranked lists, precision can be plotted against recall after each retrieved document. To facilitate comparing performance over a set of topics, each with a different number of relevant documents, individual topic precision values are interpolated to a set of standard recall levels (0 to 1 in increments of 0.1). The particular rule used to interpolate precision at standard recall level i is to use the maximum precision obtained for the topic for any actual recall level greater than or equal to i. Note that while precision is not defined at a recall 0.0, this interpolation rule does define an interpolated value for recall level 0.0. For example assume a document collection has 20 documents, four of which are relevant to topic t in which they are retrieved at ranks 1, 2, 4, 15. The exact recall points are 0.25, 0.5, 0.75, and 1.0. Using the interpolation rule, the interpolated precision for all standard recall levels 0.0, 0.1, 0.2, 0.3, 0.4, and 0.5 is 1, the interpolated precision for recall levels 0.6 and 0.7 is 0.75, and the interpolated precision for recall levels 0.8, 0.9, and 1.0 is 0.27.

3.2 Test Collection

Most information retrieval experimentation heavily depends on the existence of a test collection. A test collection is a collection of documents along with a set of test queries. The set of the relevant documents for each query is also known. To measure the recall and precision for a technique, documents are retrieved using that technique for the test queries from the collection. Since relevant documents for the queries are known, the recall and precision values can be measured. Usually the recall and precision values are averaged across all queries considered good. If a technique just works well only for a few queries, the evidence that this techniques is in general applicable is not considered strong.

As a main test collection we use TREC-7 collection (Voorhees & Harman, 1999). TREC (Text REtrieval Conference) is an DARPA (Defense Advanced Research Project Agency) and NIST (National Institute of Standards and Technology) co-sponsored effort that brings together information retrieval researchers from around the world to discuss and compare the performance of their systems, and to develop a large test collection for information retrieval system. The seventh in this series of annual conferences, TREC-7, attracted 56 different participants from academic institutions, government organizations, and commercial organizations (Voorhees & Harman, 1999). With such a large participation of various information retrieval researchers, a large and varied collections of full-text documents, a large number of user queries, and a superior set of independent relevance judgements, TREC collections have rightfully become the standard test collections for current information retrieval research.

The common information retrieval task of ranking documents for a new query is called the *adhoc* task in the TREC framework. The TREC data comes on CD-ROMs, called the TREC disks. The disks are numbered, and a combination of several disk can be used to form a text collection for experimentation.

The TREC-7 test collection consists of 50 topics (queries) and 528,155 documents from several sources: the Financial Times (FT), Federal Register (FR94), Foreign Broadcast

Information Service (FBIS) and the LA Times. Each topic consists of three sections, the *Title*, *Description* and *Narrative*. Table 2 shows statistics of the TREC-7 document collection, Table 3 shows statistics of the topics, and Figure 4 shows an example of a topic, and Figure 3 shows its expansion terms produced by our method.

It is well known that many information retrieval techniques are sensitive to factors such as query length, document length, and so forth. For example, one technique which works very well for long queries may not work well for short queries. To ensure that our techniques and conclusions are general, we use different-length query in TREC-7 collection.

Table 2: TREC-7 Document statistics

Source	Size (Mb)	Number of documents	Average number of terms/article
Disk 4			
The Financial Times, 1991-1994 (FT)	564	210,158	412.7
Federal Register, 1994 (FR94)	395	55,630	644.7
Disk 5			
Foreign Broadcast Information Services (FBIS)	470	130,471	543.6
the LA Times	475	131,896	526.5

Table 3: TREC-7 topic length statistics (words)

Topic section	Min	Max	Mean
Title	1	3	2.5
Description	5	34	14.3
Narrative	14	92	40.8
All	31	114	57.6

3.3 Baseline

For our baseline, we used SMART version 11.0 (Salton, 1971) as information retrieval engine with the *lnc.ltc* weighting method. SMART is an information retrieval engine based on the vector space model in which term weights are calculated based on term frequency, inverse document frequency and document length normalization.

Automatic indexing of a text in SMART system involves the following steps :

- **Tokenization** : The text is first tokenized into individual words and other tokens.

Title:
clothing sweatshops

Description:
Identify documents that discuss clothing sweatshops.

Narrative:
A relevant document must identify the country, the working conditions, salary, and type of clothing or shoes being produced. Relevant documents may also include the name of the business or company or the type of manufacturing, such as: "designer label".

Figure 2: Topics Example

wage	labor	sewing	low	minimum	payment
earning	workshop	workplace	shop	welfare	county
circumstance	overtime	child	entrepreneur	employment	manufacture
immigrant	industry	bussiness	company	violation	remuneration
apparel	vesture	wear	footwear	footgear	enterprise
commercialism	machine	status	plant	raise	production
calcitonin					

Figure 3: Expansion terms example

- **Stop word removal** : Common function words (like *the*, *of*, *an*, etc.) also called stop words, are removed from this list of tokens. The SMART system uses a predefined list of 571 stop words.
- **Stemming**: Various morphological variants of a word are normalized to the same stem. SMART system uses the variant of Lovin method to apply simple rules for suffix stripping.
- **Weighting** : The term (word and phrase) vector thus created for a text, is weighted using tf, idf, and length normalization considerations.

We use weighting method for document collection as follows :

$$\frac{(log(tf_{ik}) + 1.0)}{\sqrt{\sum_{j=1}^{n}[log(tf_{ij} + 1.0)]^2}}$$

and the weighting method for the initial query as follows :

$$\frac{(log(tf_{ik}) + 1.0) * log(N/n_k)}{\sqrt{\sum_{j=1}^{n}[(log(tf_{ij} + 1.0) * log(N/n_j)]^2}}$$

where tf_{ik} is the occurrrence frequency of term t_k in query q_i (for query term weighting) or in document d_i (for document term weighting), N is the total number of documents in the collection, and n_k is the number of documents to which term t_k is assigned. We use 0.1 as weight threshold (decided experimentally) and fixed for all queries.

3.4 Results

The results are shown in Table 5. This table shows the average of non-interpolated recall-precision for each of baseline, expansion using only WordNet, expansion using only predicate-argument-based thesaurus, expansion using only co-occurrence-based thesaurus, and expansion using all of them. For each method we give the percentage of improvement over the baseline. It is shown that the performance using the combined thesauri for query expansion is better than both SMART and using just one type of thesaurus.

4. Discussion

In this section we discuss why our method using WordNet is able to improve information retrieval performance. The three types of thesaurus we used have different characteristics. Automatically constructed thesauri add not only new terms but also new relationships not found in WordNet. If two terms often co-occur in a document then those two terms are likely to bear some relationship.

The reason why we should use not only automatically constructed thesauri is that some relationships may be missing in them For example, consider the words *colour* and *color*.

Table 4: TREC-7 Document statistics

Source	Size (Mb)	# Docs	Median # Words/Doc	Mean # Words/Doc
Disk 4				
FT	564	210,158	316	412.7
FR94	395	55,630	588	644.7
Disk 5				
FBIS	470	130,471	322	543.6
LA Times	475	131,896	351	526.5

Title:
clothing sweatshops

Description:
Identify documents that discuss clothing sweatshops.

Narrative:
A relevant document must identify the country, the working conditions, salary, and type of clothing or shoes being produced. Relevant documents may also include the name of the business or company or the type of manufacturing, such as: "designer label".

Figure 4: Topics Example

Table 5: TREC-7 Topic length statistics

Topic Section	Min	Max	Mean
Title	1	3	2.5
Description	5	34	14.3
Narrative	14	92	40.8
All	31	114	57.6

Table 6: Average non-interpolated precision for expansion using combined thesauri and one type of thesaurus.

Topic Type	Base	**Expanded with**			
		WordNet only	Pred-Arg only	Co-occur only	Combined
Title	0.117	0.121 (+3.6%)	0.135 (+15.2%)	0.142 (+21.2%)	0.201 (+71.7%))
Desc	0.142	0.145 (+2.5%)	0.162 (+13.1%)	0.167 (+17.3%)	0.249 (+75.3%)
All	0.197	0.201 (+1.7%)	0.212 (+7.5%)	0.217 (+10.2%)	0.265 (+34.5%)

These words certainly share the same context, but would never appear in the same document, at least not with a frequency recognized by a co-occurrence-based method. In general, different words used to describe similar concepts may never be used in the same document, and are thus missed by cooccurrence methods. However their relationship may be found in WordNet, and the syntactically-based thesaurus.

A second point is our weighting method. The advantages of our weighting method can be summarized as follows:

- the weight of each expansion term considers the similarity of that term to all terms in the original query, rather than to just one query term.
- the weight of an expansion term also depends on its similarity within all types of thesaurus.

Our method can accommodate polysemy, because an expansion term taken from a different sense to the original query term sense is given very low weight. The reason for this is that the weighting method depends on all query terms and all of the thesauri. For example, the word *bank* has many senses in WordNet. Two such senses are the financial institution and river edge senses. In a document collection relating to financial banks, the river sense of *bank* will generally not be found in the cooccurrence-based thesaurus because of a lack of articles talking about rivers. Even though (with small possibility) there may be some

```
     9. Relation. -- N. relation, bearing,  reference, connection,
concern,. cognation ; correlation c. 12; analogy; similarity c. 17;
affinity, homology, alliance, homogeneity, association; approximation c.
(nearness) 197; filiation c. (consanguinity) 11[obs3]; interest; relevancy
c. 23; dependency, relationship, relative position.
     comparison c. 464; ratio, proportion.
link, tie, bond of union.
```

Figure 5: A fragment of a Roget's Thesaurus entry

documents in the collection talking about rivers, if the query contained the finance sense of *bank* then the other terms in the query would also tend to be concerned with finance and not rivers. Thus rivers would only have a relationship with the *bank* term and there would be no relations with other terms in the original query, resulting in a low weight. Since our weighting method depends on both the query in its entirety and similarity over the three thesauri, wrong sense expansion terms are given very low weight.

5. Improving Results

5.1 Adding Roget's thesaurus as knowledge resource

Roget's Thesaurus is also a general-purpose hand-made thesaurus. In Roget's Thesaurus (Chapman, 1977), words are classified according to the ideas they express, and these categories of ideas are numbered in sequence. The terms within a category are further organized by part of speech (nouns, verbs, adjectives, adverbs, prepositions, conjunctions, and interjections). Figure 5 shows a fragment of Roget's category.

In this case, our similarity measure treat all the words in Roget as features. A word w possesses the feature f if f and w belong to the same Roget category. The similarity between two words is then defined as the Dice coefficient of the two feature vectors (Lin, 1998).

$$sim(w_1, w_2) = \frac{2|R(w_1) \cap R(w_2)|}{|R(w_1)| + |R(w_2)|}$$

where $R(w)$ is the set of words that belong to the same Roget category as w.

One may ask why we included Roget's Thesaurus here which is almost identical in nature to WordNet. The reason is to provide more evidence in the final weighting method. Including Roget's as part of the combined thesaurus is better than not including it, although the improvement is not significant (4% for title, 2% for description and 0.9% for all terms in the query). One reason is that the coverage of Roget's is very limited.

5.2 Using paragraph segmentation and Information Content for Measuring Similarity in WordNet

We tried to improve the performance by using information content for measuring the similarity between words in WordNet The similarity between words w_1 and w_2 can be defined as

Table 7: Average non-interpolated precision for expansion using single or combined thesauri.

Topic Type	Base	Expanded with				
		WordNet only	Roget only	Syntac only	Cooccur only	Combined method
Title	0.1175	0.1276 (+8.6%)	0.1236 (+5.2 %)	0.1386 (+17.9%)	0.1457 (+24.0%)	0.2314 (+96.9%)
Description	0.1428	0.1509 (+5.7%)	0.1477 (+3.4 %)	0.1648 (+15.4%)	0.1693 (+18.5%)	0.2645 (+85.2%)
All	0.1976	0.2010 (+1.7%)	0.1999 (+1.2%)	0.2131 (+7.8%)	0.2191 (+10.8%)	0.2724 (+37.8%)

the shortest path from each sense of w_1 to each sense of w_2, as below (Leacock & Chodorow, 1988) :

$$sim_{path}(w_1, w_2) = max[-\log(\frac{N_p}{2D})]$$

where N_p is the number of nodes in path p from w_1 to w_2 and D is the maximum depth of the taxonomy.

Similarity also can be measured using the information content of the concepts that subsume words in the taxonomy, as below (Resnik, 1995b) :

$$sim_{IC}(w_1, w_2) = \max_{c \in S(c_1, c_2)} [-\log p(c)]$$

where $S(c_1, c_2)$ is the set of concepts that subsume both c_1 and c_2.

Concept probabilities are computed simply as the relative frequency derived from the document collection,

$$p(c) = \frac{freq(c)}{N}$$

where N is the total number of nouns observed, excluding those not subsumed by any WordNet class.

We sum up the path-based similarity and information-content-based similarity to serve as the final similarity.

A document in TREC-7 collection can be very long and may includes multiple topics. Therefore, for improving the quality of our co-occurrence-based automatically constructed thesaurus, we change the document co-occurrence hypothesis by window co-occurrence hypothesis. We use a variable-length window-size based on the multi-paragraph topic segmentation proposed by Hearst (Hearst & Plaunt, 1993; Hearst, 1994, 1997). The main algorithm has three main parts :

- tokenization
  The text is subdivided into pseudo-sentences of a pre-defined word size s.

- similarity determination
  k pseudo-sentences are grouped together into a block to be compared against an adjacent group of pseudo-sentences (adjacent block). Similarity values are computed for every pseudo-sentence gap number; that is, score is asigned to pseudo-sentence gap i corresponding to how similar the pseudo-sentences from pseudo-sentence $i-k$ through i are to the pseudo-sentence from $i+1$ to $i+k+1$. Similarity between blocks is calculated by a cosine measure: given two text blocks $b1$ and $b2$, each with k pseudo-sentences,
  $$sim(b1, b2) = \frac{\sum_t w_{t,b1} w_{t,b2}}{\sqrt{\sum_t w_{t,b1}^2 \sum_t w_{t,b2}^2}}$$
  where t ranges over all the terms that have been registered during the tokenization, and $w_{t,b1}$ is their frequency within the $b1$ block.

- Boundary identification
  Boundaries are determined by changes in the sequence of similarity scores. For a given pseudo-sentence gap i, the algorithm looks at the scores of the pseudo-sentence gaps to the left of i as long as their values are increasing. When the values to the left peak out, the difference between the score at the peak and the score at i is recorded. The same procedure is performed with the pseudo-sentence gaps to the right of i. Finally, the relative height of the peak to the right of i is added to the relative height of the peak to the left. These new scores, called $depth - scores$, correspond to how sharp a change occurs on both sides of the pseudo-sentence gap. After performing average smoothing, a boundary is determined by defining the cutoff as a function of the average and standard deviation of the $depth - scores$ for the text.

In this paper, we used the parameter as belows :

- Width of the pseudo-sentences (s) is 20
- Blocksize (k) is 6

After the topic-segments are determined, we use mutual information as a tool for computing similarity between words. Mutual information compares the probability of the co-occurence of words a and b with the independent probabilities of occurrence of a and b :

$$I(a, b) = log \frac{P(a, b)}{P(a)P(b)}$$

where the probabilities of $P(a)$ and $P(b)$ are estimated by counting the number of occurrences of a and b in topic-segments. The joint probability is estimated by counting the number of times that word a co-occurs with b.

Table 8 shows the average of 11-point interpolated precision using various section of topics in TREC-7 collection, and Table 11 shows the average of 11-point interpolated precision in several small collections. We can see that our method give a consistent and significant improvement compared with the baseline and using only one type of thesaurus.

Table 8: Experiment results using TREC-7 Collection

Topic Type	Base	Expanded with						
		WordNet only	Syntactic only	Cooccur only	WordNet+ Syntactic	WordNet+ Cooccur	Syntactic+ Cooccur	Combined method
Title	0.1452	0.1541 (+6.1%)	0.1802 (+24.1%)	0.1905 (+31.2%)	0.1877 (+29.3%)	0.2063 (+42.1%)	0.2197 (+51.3%)	0.2659 (+83.1 %)
Description	0.1696	0.1777 (+4.8%)	0.1974 (+16.4%)	0.2144 (+26.4%)	0.2057 (+21.3%)	0.2173 (+28.1%)	0.2337 (+37.8%)	0.2722 (+60.5 %)
All	0.2189	0.2235 (+2.1%)	0.2447 (+11.8%)	0.2566 (+17.2%)	0.2563 (+17.1%)	0.2611 (+19.3%)	0.2679 (+22.4%)	0.2872 (+31.2 %)

6. Experiment using Small and Domain-dependent Test Collection

Beside the large and the newer TREC-7 test collection described before, we also use some previous small test collections (Fox, 1990), because although most real world collections are large, some can be quite small. These small collections have been widely used in the experiments by many information retrieval researchers before TREC. These old test collections have always been built to serve some purpose. For example, the Cranfield collection was originally built to test different types of manual indexing, the MEDLINE collection was built in an early attempt to compare the operational Boolean MEDLARS system with the experimental ranking used in SMART, and the CACM and CISI collections were built to investigate the use of an extended vector space model that included bibliographic data. Most of the old test collections are very domain specific and contain only the abstract.

In Table 9 and 10 we describe the statistics and the domain of the old collection, respectively.

Table 9: Small collection statistics

Collection	Number of Documents	Average Terms/Docs	Number of Query	Average Terms/query	Average Relevant/query
Cranfield	1398	53.1	225	9.2	7.2
ADI	82	27.1	35	14.6	9.5
MEDLARS	1033	51.6	30	10.1	23.2
CACM	3204	24.5	64	10.8	15.3
CISI	1460	46.5	112	28.3	49.8
NPL	11429	20.0	100	7.2	22.4
INSPEC	12684	32.5	84	15.6	33.0

7. Failure Analysis

Although our method as a whole gives a very significant improvement, it still further can be improved. Of the 50 queries of TREC-7 collection, our method improves the performance of 43 queries and degrade the performance of 7 queries compared with the baseline. We investigated manually why our method degrade the performance of several queries.

Table 10: The domain of the small collections

Collection	Domain
Cranfield	Aeronautics
ADI	Information Science
MEDLINE	Medical Science
CACM	Computer Science
CISI	Computer and Information Science
NPL	Electrical Engineering
INSPEC	Electrical Engineering

Table 11: Experiment results using small collection

Coll	Base	Expanded with						
		WordNet only	Syntactic only	Cooccur only	WordNet+ Syntactic	WordNet+ Cooccur	Syntactic+ Cooccur	Combined method
ADI	0.4653	0.4751 (+2.1%)	0.5039 (+8.3%)	0.5146 (+10.6%)	0.5263 (+13.1%)	0.5486 (+17.9%)	0.5895 (+26.7%)	0.6570 (+41.2%)
CACM	0.3558	0.3718 (+4.5%)	0.3853 (+8.3%)	0.4433 (+24.6%)	0.4109 (+15.5%)	0.4490 (+26.2%)	0.4796 (+34.8%)	0.5497 (+54.5%)
INSPEC	0.3119	0.3234 (+3.7%)	0.3378 (+8.3%)	0.3755 (+20.4%)	0.3465 (+11.1%)	0.4002 (+28.3%)	0.4420 (+41.7%)	0.5056 (+62.1 %)
CISI	0.2536	0.2719 (+7.2%)	0.2800 (+10.4%)	0.3261 (+28.6%)	0.3076 (+21.3%)	0.3606 (+42.2%)	0.4009 (+58.1%)	0.4395 (+73.3 %)
CRAN	0.4594	0.4700 (+2.3%)	0.4916 (+7.0%)	0.5435 (+18.3%)	0.5012 (+9.1%)	0.5706 (+24.2%)	0.5931 (+29.1%)	0.6528 (+42.1 %)
MEDLINE	0.5614	0.5681 (+1.2%)	0.6013 (+7.1%)	0.6372 (+13.5%)	0.6114 (+8.9%)	0.6580 (+17.2%)	0.6860 (+22.2%)	0.7551 (+34.5%)
NPL	0.2700	0.2840 (+5.2%)	0.2946 (+9.1%)	0.3307 (+22.5%)	0.3038 (+12.5%)	0.3502 (+29.7%)	0.3796 (+40.6%)	0.4469 (+65.5%)

7.1 Negation statements in the query

We found that most of the queries hurted by our method contains the negation statements. Through our method, all the terms in the negation statements are also considered for query expansion which is degrading the retrieval performance for that query. Figure 6 shows two examples of query which contain negation statements.

Table 12 shows the results of eliminating the negation statements from the queries manually for each query containing negation statements. As that table shown, eliminating the negation statements improves the retrieval effectiveness. It is to be investigated further how we could identify the negation statements automatically.

7.2 Multiple aspects of query

An examination of the top-ranked non-relevant documents for various queries shows that a commonly occurring cause of non-relevance among such documents is inadequate query coverage, i.e., the query consists of multiple aspects, only some of which are covered in these documents. For example, a query of the TREC collection asks : *Identify documents discussing the use of estrogen by postmenopausal women in Britain.* Several top-ranked non-relevant documents contain information about the use of hormone by postmenopausal

Title:
British Chunnel impact

Description:
What impact has the Chunnel had on the British economy and/or the life style
of the British?

Narrative:
Documents discussing the following issues are relevant:
- projected and actual impact on the life styles of the British
- Long term changes to economic policy and relations
- major changes to other transportation systems linked with the Continent
Documents discussing the following issues are not relevant:
- expense and construction schedule
- routine marketing ploys by other channel crossers (i.e., schedule changes, price
drops, etc.)

Title:
Ocean remote sensing

Description:
Identify documents discussing the development and application of spaceborne ocean remote sensing.

Narrative:
Documents discussing the development and application of spaceborne ocean remote sensing in oceanography, seabed prospecting and mining, or any marine-science activity are relevant. Documents that discuss the application of satellite remote sensing in geography, agriculture, forestry, mining and mineral prospecting or any land-bound science are not relevant, nor are references to international marketing or promotional advertizing of any remote-sensing technology. Synthetic aperture radar (SAR) employed in ocean remote sensing is relevant.

Figure 6: Two examples of query containing negation statements

Table 12: The results of negation statements elimination

Query Number	SMART	Expansion without Negation Elimination	Expansion with Negation Elimination
2	0.3643	0.3381 (- 7.19%)	0.3811 (+ 4.61%)
5	0.3112	0.2804 (- 9.90%)	0.3314 (+ 6.49%)
13	0.1621	0.1567 (- 3.33%)	0.1823 (+12.46%)
17	0.2310	0.2235 (- 3.25%)	0.2441 (+ 5.67%)
42	0.2732	0.2569 (- 5.97%)	0.2942 (+ 7.69%)
43	0.3031	0.2834 (- 6.50%)	0.3321 (+ 9.57%)

women but not in Britain. If we look at the expansion terms produced by our method as shown in Figure 7 we could see that many expansion terms have relationship with all query terms except *Britain*. This is because all query terms but *Britain* have relationship between each other and these terms have a high original term weight. On the contrary, *Britain* does not have relationship with other query terms and *Britain* have a low original term weight in almost all documents in collection. Consequently, the term related to *Britain* are given a low weight by our method.

estradiol	female	hormone	disease	therapy	menopausal
chemical	progesterone	menstruation	vaginal	progestin	obstetrics
gynecology	replacement	endometrial	cancer	breast	ovary
treatment	old	tamoxifen	symptom	synthetic	drug
hot	flash	osteoporosis	cholesterol	recepter	risk
calcium	bones	mineralization	medical	physiologist	diagnostic
calcitonin					

Figure 7: Expansion terms

To investigate the relatedness or independence of query words, we examine their co-occurrence patterns in 1000 documents initially retrieved for a query. If two words have the same aspect, then they often occur together in many of these documents. If one of the words appears in a document, the chance of the other occurring within the same document is likely to be relatively high. On the other hand, if two words bear independent concepts, the occurrences of the words are not strongly related.

Based on this observation, we re-rank the top-1000 retrieved documents, by re-computing the similarity between a query $\overrightarrow{q} = \{t_1, t_2, .., t_m\}$ (terms are ordered by decreasing of their inverse document frequency) and document D as belows (Mitra, Singhal, & Buckley, 1998) :

$$Sim_{new}(D) = idf(t_1) + \sum_{i=2}^{m} idf(t_i) \ \times \ min_{j=1}^{i-1}(1 - P(t_i|t_j)),$$

where *idf* is the inverse of document frequency in the top-1000 initially retrieved documents, m is the number of terms in query that appear in document D, and $P(t_i|t_j)$ is estimated based on word occurrences in document collection and is given by :

$$\frac{\#\ documents\ containing\ words\ t_i\ and\ t_j}{\#\ documents\ containing\ word\ t_j}.$$

For example, in the query stated above, the terms *estrogen, postmenopausal,* and *women* are strongly related to each other. If the term *postmenopausal* occurs in a document, the probability of word *women* occurring in the same document is high. Accordingly, the contribution of word *women* to Sim_{new} is reduced in this case. On the other hand, terms *postmenopausal* and *Britain* correspond to two independent aspects of the query and the occurrences of these two terms are relatively uncorrelated. Therefore, if a document contains these two terms, the contribution of *Britain* is higher and it counts as an important new matching term since its occurrence is not well predicted by other matching term (*postmenopausal*). This technique can improve the average of 11-point interpolated precision of TREC-7 collection for about 3.3% as shown in Table 13.

We also investigated another method to overcome this problem in which we built a Boolean expression for all query manually. Terms in the same aspect of query are placed in *or* relation, and terms in different aspect are placed in *and* relation (Hearst, 1996). Documents that satisfy the constraint contain at least one word from each aspect of the query. For example, for the query stated before (*Identify documents discussing the use of estrogen by postmenopausal women in Britain*), we construct boolean expression as follows:

```
estrogen and (postmenopausal or woman) and britain.
```

Using this method, we again re-rank the top 1000 documents initially retrieved. Documents that match more words in different aspect of query are ranked ahead of documents that match less words. Ties are resolved by referring to the original document weight. Using this method we can improve the average of 11-point interpolated precision of TREC-7 collection for about 11.3%, as shown in Table 13.

This correlation and boolean reranking methods degrade some queries performance, because in those queries these methods overweight several query terms.

It is to be further investigated how we could design the appropriate method to overcome this problem.

8. Combining with relevance feedback

In this section, we describe the combination of our method with pseudo-relevance feedback (Buckley & Salton, 1994, 1995; Salton & Buckley, 1990). Pseudo-relevance feedback is a

Table 13: The effect of re-ranking the top-1000 ranked initially retrieved using co-occurrence method and boolean filter method

Query Number	Without Re-ranking	Re-ranking correlation	%improvement	Reranking Boolean	%improvement
1	0.5153	0.5666	+9.96	0.7724	+49.89
2	0.3794	0.1952	-48.55	0.4740	+24.93
3	0.3230	0.2719	-15.82	0.3237	+0.22
4	0.2280	0.2731	+19.78	0.2355	+3.29
5	0.3213	0.2457	-23.53	0.2931	-8.78
6	0.0646	0.0495	-23.37	0.0655	+1.39
7	0.3878	0.5632	+45.23	0.3607	-6.99
8	0.2983	0.4270	+43.14	0.3049	+2.21
9	0.0422	0.0612	+45.02	0.0254	-39.81
10	0.2196	0.3223	+46.77	0.3619	+64.80
11	0.5802	0.3524	-39.26	0.4950	-14.68
12	0.3588	0.1466	-59.14	0.2319	-35.37
13	0.1745	0.0908	-47.97	0.0868	-50.26
14	0.6055	0.5604	-7.45	0.4963	-18.03
15	0.8877	0.9451	+6.47	0.8554	-3.64
16	0.3856	0.3094	-19.76	0.4823	+25.08
17	0.2360	0.1363	-42.25	0.1479	-37.33
18	0.7882	0.6419	-18.56	0.6662	-15.48
19	0.5141	0.4027	-21.67	0.4177	-18.75
20	0.1871	0.3997	+113.63	0.3016	+61.20
21	0.0152	0.0346	+127.63	0.0837	+450.66
22	0.0920	0.3644	+296.09	0.1399	+52.07
23	0.2328	0.4043	+73.67	0.4277	+83.72
24	0.3250	0.3177	-2.25	0.3951	+21.57
25	0.5943	0.2812	-52.68	0.3239	-45.50
26	0.2360	0.2312	-2.03	0.1034	-56.19
27	0.4634	0.3062	-33.92	0.3322	-28.31
28	0.0307	0.0306	-0.33	0.0142	-53.75
29	0.0314	0.2575	+720.06	0.3349	+966.56
30	0.2162	0.2164	+0.09	0.3832	+77.24
31	0.0500	0.0560	+12.00	0.0635	+27.00
32	0.4544	0.5968	+31.34	0.5803	+27.71
33	0.0220	0.0232	+5.45	0.0290	+31.82
34	0.2169	0.1989	-8.30	0.2299	+ 5.99
35	0.2267	0.3421	+50.90	0.4012	+76.97
36	0.0129	0.0286	+121.71	0.0406	+214.73
37	0.2563	0.2605	+1.64	0.2289	-10.69
38	0.2534	0.2300	-9.23	0.2079	-17.96
39	0.0006	0.0200	+3233.33	0.0085	+1316.67
40	0.2004	0.3230	+61.18	0.2708	+35.13
41	0.0015	0.4938	+32820.00	0.5261	+34973.33
42	0.2883	0.1346	-53.31	0.4216	+46.24
43	0.2996	0.1280	-57.28	0.1684	-43.79
44	0.0218	0.1019	+367.43	0.0952	+336.70
45	0.1506	0.1879	+24.77	0.2783	+84.79
46	0.3485	0.6087	+74.66	0.4719	+35.41
47	0.0967	0.0303	-68.67	0.3293	+240.54
48	0.3886	0.3418	-12.04	0.2954	-23.98
49	0.2066	0.1351	-34.61	0.1826	-11.62
50	0.3861	0.4312	+11.68	0.3978	+3.03
Average	0.2723	0.2815	+3.3	0.3033	+11.3

feedback aproach without requiring relevance information. Instead, an initial retrieval is performed, and the top-n ranked documents are all assumed to be relevant for obtaining expansion terms ($\overrightarrow{\mathrm{q}}_{feedback}$) as belows :

$$\overrightarrow{\mathrm{q}}_{feedback} = \frac{1}{|D_r|} \sum_{d_i \in D_r} \overrightarrow{d_i}$$

In this case, D_r is a set of documents ranked on the top in the initial retrieval and $\overrightarrow{d_i}$ is the vector representation of document d_i.

In the framework of the inference network (Xu & Croft, 1996), the information need of the user is represented by multiple queries. Multiple queries means that an information need is represented by some different query representation. Experiments show that multiple query representations can produce better results than using one representation alone. However, how to obtain these queries is not discussed in this model. Hence we try to find multiple query representations for the information structure derived from feedback information. In this way, the following three representations can be obtained :

- representation derived directly from the original query : $\overrightarrow{\mathrm{q}}_{original}$,
- representation obtained by our method : $\overrightarrow{\mathrm{q}}_{thesauri}$,
- representation derived from the retrieved documents of the previous run : $\overrightarrow{\mathrm{q}}_{feedback}$.

A linear combination of the three query representations is used to retrieve documents. However, we do not introduce additional parameters which are quite difficult to determine. Also we believe that the parameter values determined for some queries may not be suitable for some other queries because they are query dependent. Hence the simple combination we use is :

$$\overrightarrow{\mathrm{q}}_{original} + \overrightarrow{\mathrm{q}}_{thesauri} + \overrightarrow{\mathrm{q}}_{feedback}.$$

When using the relevance-feedback method, we used the top 30 ranked documents of the previous run of the original query to obtain $\overrightarrow{\mathrm{q}}_{feedback}$.

In order to evaluate the retrieval effectiveness of the new method, we carried out some experiments using TREC-7 collection to compare the retrieval effectiveness of the following methods using different combination of the query representations. Figure 8 shows 11-point interpolated precision using our method alone, pseudo-feedback alone, and the combination of our method and pseudo-feedback. Our method alone has better performance than the pseudo-feedback method, and the combination of our method and pseudo-feedback slightly better than our method alone.

Recently, Xu and Croft (1996) suggested a method called local context analysis, which also utilize the co-occurrence-based thesaurus and relevance feedback method. Instead of gathering co-occurrence data from the whole corpus, he gather it from the top-n ranked document. We carry out experiments in that we build the combined-thesauri based on the top-n ranked document, rather than the whole corpus. As can be seen in Figure 9, query expansion using the combined thesauri built from the top-n ranked document have a lower performance than query expansion using the combined thesauri built from the whole corpus.

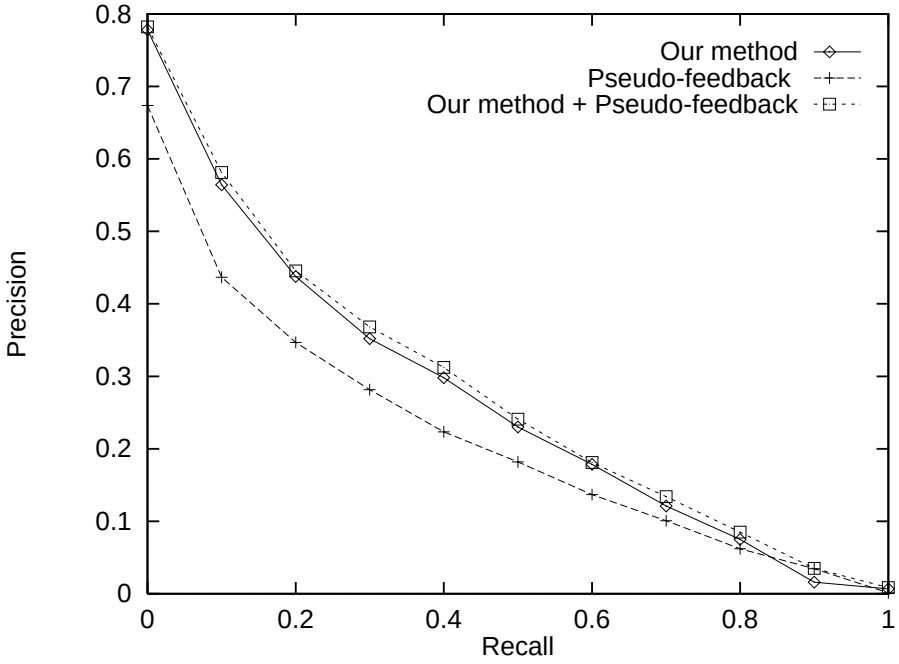

Figure 8: The results of combining our method and pseudo-feedback

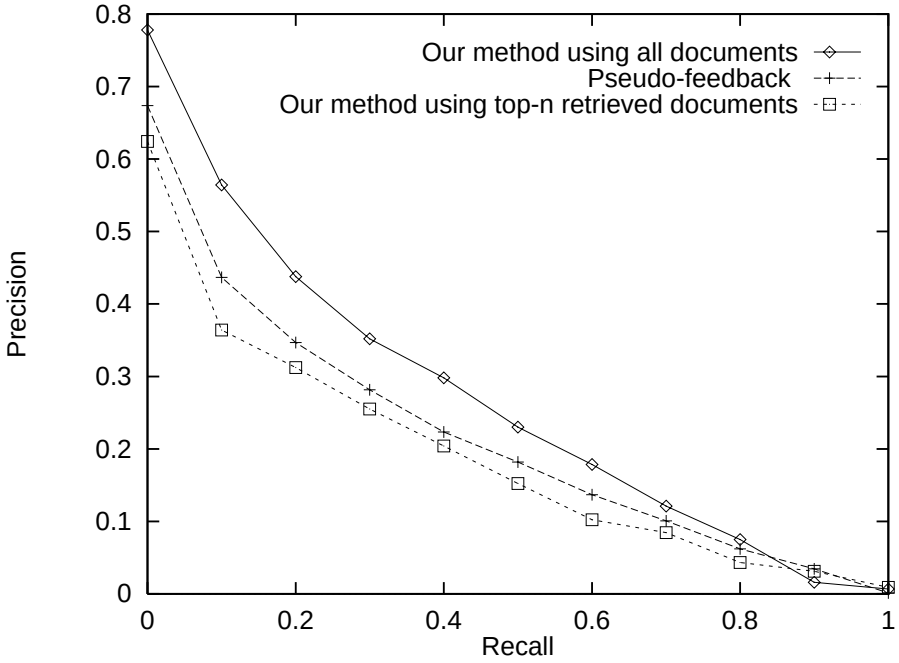

Figure 9: The results of combined thesauri built from the top-n ranked document

9. Conclusions and Future Work

We have proposed the use of multiple types of thesauri for query expansion in information retrieval, give some failure analysis, and combining our method with pseudo-relevance feedback method. The basic idea underlying our method is that each type of thesaurus has different characteristics and combining them provides a valuable resource to expand the query. Misleading expansion terms can be avoided by designing a weighting term method in which the weight of expansion terms not only depends on all query terms, but also depends on their similarity values in all type of thesaurus.

Future research will include the use of parser with better performance, designing a general algorithm for automatically handling the negation statements, and also designing an effective algorithm for handling the multiple aspect contain in the query.

References

Buckley, C. & Salton, G. (1994). The Effect of Adding Relevance Information in a Relevance Feedback Environment. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval Conference*, pp. 292–300.

Buckley, C. & Salton, G. (1995). Automatic Query Expansion using SMART: TREC-3. In *Proceedings of The Third Text Retrieval Conference*, pp. 69–80.

Chai, J. & Biermann, A. (1997). The use of Lexical Semantics in Information Extraction. In *Proceedings of the ACL-EACL Workshop on Automatic Information Extraction and Building of Lexical Semantic Resources*, pp. 61–70.

Chapman, R. L. (1977). *Roget's International Thesaurus (4th Edition)*. Harper and Row, New York.

Fox, E. A. (1990). *Virginia Disk One*. Blacksburg: Virginia Polytechnic Institute and State University.

Gomez-Hidalgo, J. & Rodriguez, M. (1997). Integrating a Lexical Database and a Training Collection for Text Categorization. In *Proceedings of the ACL-EACL Workshop on Automatic Information Extraction and Building of Lexical Semantic Resources*, pp. 39–44.

Grefenstette, G. (1992). Use of Syntactic Context to Produce Term Association Lists for Text Retrieval. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 89–97.

Grefenstette, G. (1994). *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publisher.

Hearst, M. A. (1994). Multi-Paragraph Segmentation of Expository Text. In *Proceedings of 32th Annual Meeting of the Association for Computational Linguistics*, pp. 9–16.

Hearst, M. A. (1996). Improving Full-Text Precision on Short Queries using Simple Constraints. In *Proceedings of the 5th Annual Symposium on Document Analysis and Information Retrieval (SDAIR)*.

Hearst, M. A. (1997). TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. *Computational Linguistics*, *23*(1), 33–64.

Hearst, M. A. & Plaunt, C. (1993). Subtopic Structuring for full-length document access. In *Proceedings of 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 59–68.

Hindle, D. (1990). Noun Classification from Predicate-Argument Structures. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistic*, pp. 268–275.

Leacock, C. & Chodorow, M. (1988). Combining Local Context and WordNet Similarity for Word Sense Identification. In Fellbaum, C. (Ed.), *WordNet,An Electronic Lexical Database*, pp. 265–283. MIT Press.

Lin, D. (1998). Automatic Retrieval and Clustering of Similar Words. In *Proceedings of the COLING-ACL'98*, pp. 768–773.

Miller, G. A. (1990). Special Issue, WordNet: An On-line Lexical Database. *International Journal of Lexicography*, *3*(4).

Mitra, M., Singhal, A., & Buckley, C. (1998). Improving Automatic Query Expansion. In *Proceedings of the 21th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*, pp. 206–214.

Morris, J. & Hirst, G. (1991). Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pp. 21–45.

Qiu, Y. & Frei, H.-P. (1993). Concept Based Query Expansion. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 160–169.

Resnik, P. (1995a). Disambiguating Noun Grouping with Respect to WordNet Senses. In *Proceedings of 3rd Workshop on Very Large Corpora*.

Resnik, P. (1995b). Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, pp. 448–453.

Richardson, R. & Smeaton, A. F. (1995). Using WordNet in a Knowledge-Based Approach to Information Retrieval. Tech. rep. CA-0395, School of Computer Applications, Dublin City University.

Ruge, G. (1992). Experiments on Linguistically-Based Term Associations. *Information Processing and Management*, *28*(3), 317–332.

Salton, G. & Buckley, C. (1990). Improving Retrieval Performance by Relevance Feedback. *Journal of American Society for Information Science*, *41*(4), 288–297.

Salton, G. (1971). *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice-Hall.

Segond, F., Schiller, A., Grefenstette, G., & Chanod, J. (97). An Experiment in Semantic Tagging using Hidden Markov Model Tagging. In *Proceedings of the ACL-EACL Workshop on Automatic Information Extraction and Building of Lexical Semantic Resources*, pp. 78–81.

Sekine, S. & Grishman, R. (1995). A Corpus-based Probabilistic Grammar with Only Two Non-terminals. In *Proceedings of the International Workshop on Parsing Technologies*.

Smeaton, A. F. & Berrut, C. (1995). Running TREC-4 Experiments: A Chronological Report of Query Expansion Experiments Carried out as Part of TREC-4. In *Proceedings of The Fourth Text REtrieval Conference (TREC-4)*. NIST special publication.

Stairmand, M. A. (1997). Textual Context Analysis for Information Retrieval. In *Proceedings of the 20th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 140–147.

Voorhees, E. M. (1993). Using WordNet to Disambiguate Word Senses for Text Retrieval. In *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 171–180.

Voorhees, E. M. (1994). Query Expansion using Lexical-Semantic Relations. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 61–69.

Voorhees, E. & Harman, D. (1999). Overview of the Seventh Text retrieval Conference (TREC-7). In *Proceedings of the Seventh Text REtrieval Conference*. NIST Special Publication.

Xu, J. & Croft, B. (1996). Query Expansion Using Local and Global Document Analysis. In *Proceedings of the 19th ACM-SIGIR Conference*, pp. 4–11.